

Definition

Term: Closures (A term from CPEN400A class)

Parenthetical Definition:

The function implements a closure (a record storing function environment) inside itself.

Sentence Definition:

Closure is a programming technique implemented in programming languages (especially in JavaScript). This technique is usually implementing as a nested function storing enclosing function with its environment regardless whether enclosing function exists.

Expanded Definition:

The term Closure was defined by Peter J. Landin in 1964 when he was using SECD machine for evaluating expressions. Nowadays, we use this term refer to those nested function which access parent's function to capture variables through its closure' scope. Regardless the old function returned or no longer exists, you can preserve all the elements of state that the inner function refers to.

A simple closure example consists of a enclosing function return a child function. The basic framework displayed in the Code below. Inside the parent function, variables, object, other functions referred by closure could be stored in an unmanipulated place in order to protect the data from accidentally change. The return function can be nested inside an Object because every function in JavaScript is Object. For example here below is a typical JavaScript closure.

```
function Counter(initial) {  
  var val = initial;
```

```
return {
  increment : function() { val += 1; },
  reset: function() { val = initial; },
  get: function() { return val; }
};
};
```

Why Closures are useful?

Closures allow program to remember state in Web Applications especially when you have many different handlers construct parts of an object. Besides, when AJAX working for transition, closures are very useful for callbacks in JavaScript by returning the callback function from the parent function. In addition, it is a way to emulate private variables in complexed working environment.

What do you need to be careful of?

Nested functions cannot access the parent function's this and arguments because "this" is always a function local reference that is set by the caller of the function.